Demonstration of Autonomous Orbit Determination Around Small Bodies

S. Bhaskaran, J. E. Riedel, and S. P. Synnott
Navigation and Flight Mechanics Section
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

Abstract

An important component of future space missions is to examine small solar system objects such as asteroids with inexpensive spacecraft. In order to reduce the costs of such missions, it is desirable to perform some or all of the navigation function onboard the spacecraft. This paper presents an algorithm to autonomously determine the orbit of the spacecraft using a wide field-of-view camera and a precomputed model of the object. In the algorithm, the observed limbs of the object are compared with its precomputed location using the model to obtain an instantaneous position fix of the spacecraft. A series of these position fixes are then input to a filter which determines the complete state of the spacecraft. The procedure is validated using simulations of orbits around sample asteroids.

1 Introduction

The navigation]) of spacecraft orbiting small objects such as asteroids presents special problems not seen in planetary orbiting missions. In particular, the dynamic environment the spacecraft is operating in may be changing quickly due to perturbations from the gravity field and solar radiation pressure. Maneuvers are then needed fairly often to maintain the orbit. Navigating the spacecraft using colRel;-tional gyound-based methods may be very costly in terms of staffing requirements to keep personnel around the clock to perform orbit determination and maneuver analysis functions. It may also prove to be impractical to navigate from the ground due to frequent maneuvers and long round-trip light times. For these reasons, the feasibility of performing some or all of the navigation functions onboard the spacecraft is being looked into. This study addresses the problem of onboard orbit determination of a spacecraft around a small object. It is assured that an important first step to any autonomous scheme for orbiting small objects - building a model of the object itself - has been done and is available. An algorithm has been developed which uses this precomputed shape model of the object and images taken with awidcfield-of-view (FOV) camera to autonomously determine the state of the spacecraft. The method is demonstrated with the use of a simulation.

2 Equipment

The sole instrument used to obtain data is a camera with a wide FOV (around 50 to 60 degrees) and a fairly short focal length (50 to 100 mm). The exact specifications of the camera will depend on the particular mission; for this simulation, an 800x800 pixel array camera with a 60 deg. FOV and 50 mm

focal length was chosen. The wide FOV is necessary to keep the entire object within the camera frame even at fairly close distances. In addition, these specifications ensure that epected errors in camera pointing obtained independently of the imaging camera will be at the sub-pixel level and can be ignored for the simulation.

3 Image Processing

3.1 Object Modeling

It is assumed in the subsequent description of the image processing methodology that a reasonably good model of the object being orbited, including its spin rate and orientation in space, is available. Although the details of how this is obtained is beyond the scope of this text, a brief discussion of the object modeling process is necessary.

First, a suitable format for the objects shape model is needed. Although several formats exist, such as spherical harmonic expansions of the topography and albedo, or interconnected flat plates (Ref. 1), we chose a new representation which is both compact and easy to represent unusual shapes. The basic idea is to represent the object as a three-dimensional matrix. In each block of the matrix, a number is used to describe whether the Mock is occupied by the object, or is empty space. Thus, the object is built up as a set of occupied blocks in the 3-D matrix.

With the representation chosen, the second problem is to build the model from pictures of the object. Unfortunately, unless the mission is to Ida or Gaspra, no pictures arc available of any asteroids. One possible solution if the asteroid is a near Earth one is to obtain a rough estimate of the shape from radar bounces off the object, such as was done for Castalia (Ref. 2). The rotation rate can be obtained from examining the light curve from ground based observations, but the orientation may be problematic. The second option, therefore, is to obtain the shape model from picture taken by the spacecraft in the vicinity of the asteroid. In a mission to an asteroid, the nominal mission plan would undoubtedly include a phase where the spacecraft would be either be very slowly approaching the object, or in a loosely bound orbit at a great distance from the center. During this period, the on board camera would be taking pictures of the asteroid in order to characterize its properties, such as its sire, shape, orientation, and spin characteristics. From this observation campaign, which might last weeks or even months, a low order shape model can be obtained, as well as accurate estimates of its pole orientation and rotation period. Whether this can be done autonomously onboard, or whether it must be done 011 the ground is still open to question. Currently, software exists in the Optical Systems Analysis group at JPL which take.s in a set of pictures of an object and computes a shape model using spherical harmonics (Ref. 3). In addition, an effort is underway to refine this procedure to produce a model in the block format described above. Future efforts will include automating this process for use in an onboard system.

3.2 Spacecraft Point Positioning

Historically, optical navigation has been used primarily to obtain target relative angular measurements which supplemented standard radio navigation techniques for deep-space missions (Ref. 4). To obtain the measurement, techniques were developed to perform high precision centerfinding (0.1 pixel or less) on objects which were unresolved point sources or ellipsoidal extended bodies. For orbiting an asteroid whose shape model is available, however, an additional observable, namely the range to the object, can be inferred from the image. In particular, the shape model enables a deterministic measurement of the spacecraft's object centered position from triangulation. This process is easily visualized from Figure 1;

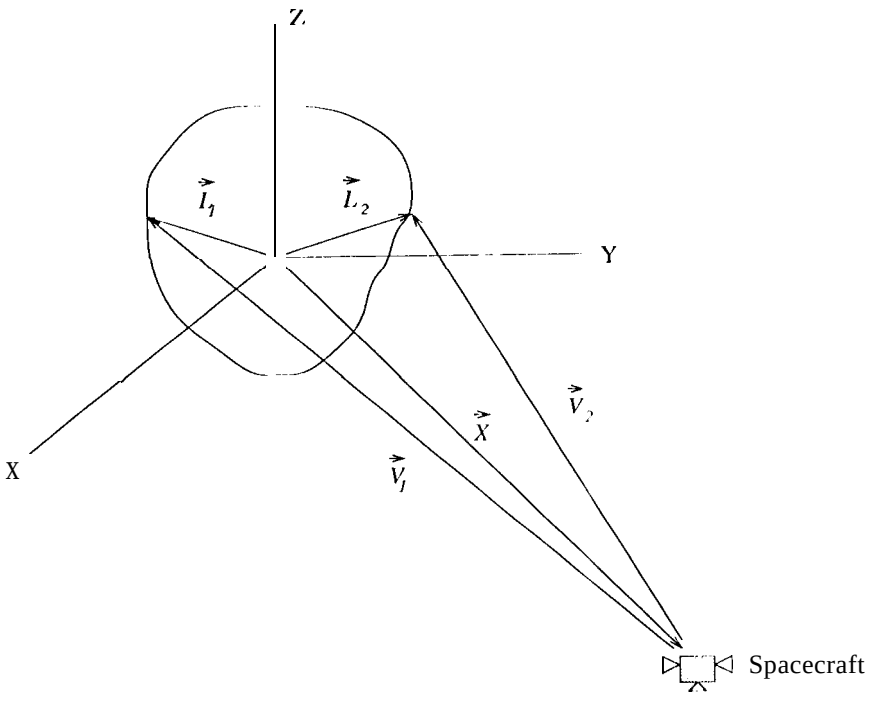

Figure 1: Spacecraft Viewing Geometry

the spacecraft lies on the intersection of two line-of-sight vectors ($\vec{V}_1$ and $\vec{V}_2$) to two known limbs ($\vec{L}_1$ and $\vec{L}_2$) of the object. In principle, ibis measurement is sufficient as an instantaneous position fix for subsequent incorporation into a filter for full spacecraft state estimation. 1 n practice, however, it is both necessary and desirable to get a filtered position fix as well due to inaccuracies in the object model and limb correlation techniques. in addition, since a nominal trajectory is available, the problem can be set up as a linearized least-squares estimate about the nominal position, which greatly simplifies the computation. The filtered position fix also provides formal statistics of of the position estimate for usc in the full state filter.

The P rocess to obtain a filtered estimate of the instantaneous position will now be described. In order to simplify the equations, the assumption is made that camera distortions are ignored and that the camera boresight is along the spacecraft $z-$axis (see Ref. 5 for a more detailed derivation). Also, it is assumed that limb vectors have been rotated into a body-fixed inertial reference frame. From Figure 1, it can be seen that the spacecraft's inertial body-centered position vector is $\vec{X}$, the variable to be computed. The observed limb in the camera image is the line-of-sight vector, $\vec{V}$ projected into the camera focal plane, where $\vec{V}$ can be computed as $\vec{L}-\vec{X}$. To transform $\vec{V}$ into pixel and line coordinates, first rotate $\vec{V}$ into the camera frame (a frame with the $z-$ axis along the camera boresight, and the x and y axes determined by the camera twist angle):

$$\vec{V}_c = T_{IC}\vec{V} \tag{1}$$

Where

V_c = the line-of-sight vector to the limb in the camera frame, and

T_{IC} = the rotation matrix from inertial to the camera frame.

Then, project $\vec{V}_c$ into the camera focal plane via the lens equation:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{V_{c3}} \begin{bmatrix} V_{c1} \\ V_{c2} \end{bmatrix} \tag{2}$$

where

f = the camera focal length, in mm
x, y = the limb location in focal plane coordinates, in mm, and
V_{c1}, V_{c2}, V_{c3} = the components of the line-of-sight vector.

Finally, the limb location in pixel and line coordinates p, and 1 is

$$\begin{bmatrix} p \\ l \end{bmatrix} = K \begin{bmatrix} x \\ y \end{bmatrix} \tag{3}$$

where

K = the conversion matrix to go from mm to pixel and line.

To perform the least-squares estimate, partial derivatives of the observable (the pixel and line coordinate of the limb) with respect to the instantaneous position of the spacecraft are needed. These partials are obtained by first differentiating the limb point in focal plane mm coordinates with the line-of-sight vector $\vec{V}$ (Ref. 5):

$$\begin{bmatrix} \frac{\partial x}{\partial \vec{V}} \\ \frac{\partial y}{\partial \vec{V}} \end{bmatrix} = \frac{\frac{\partial f}{\partial \vec{V}} V_{c3} - f \frac{\partial V_{c3}}{\partial \vec{V}}}{V_{c3}^2} \begin{bmatrix} V_{c1} \\ V_{c2} \end{bmatrix} + \frac{f}{V_{c3}} \begin{bmatrix} \frac{\partial V_{c1}}{\partial \vec{V}} \\ \frac{\partial V_{c2}}{\partial \vec{V}} \end{bmatrix} \tag{4}$$

Note that $\frac{\partial f}{\partial \vec{V}}$ is zero if the camera focal length is constant, and that $\frac{\partial \vec{V}_c}{\partial \vec{V}}$ is simply T_{IC}, the inertial to camera frame rotation matrix. Then, the pixel and line partials are:

$$\begin{bmatrix} \frac{\partial p}{\partial \vec{V}} \\ \frac{\partial l}{\partial \vec{V}} \end{bmatrix} = \begin{bmatrix} \frac{\partial p}{\partial y} \\ \frac{\partial l}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial x}{\partial \vec{V}} \\ \frac{\partial y}{\partial \vec{V}} \end{bmatrix} = K \begin{bmatrix} \frac{\partial x}{\partial \vec{V}} \\ \frac{\partial y}{\partial \vec{V}} \end{bmatrix} \tag{5}$$

The last step is to perform the chain rule to go from partials with respect to the line-of-sight vector to partials with respect to spacecraft position:

$$\begin{bmatrix} \frac{\partial p}{\partial \vec{X}} \\ \frac{\partial l}{\partial \vec{X}} \end{bmatrix} = \begin{bmatrix} \frac{\partial p}{\partial \vec{V}} \\ \frac{\partial l}{\partial \vec{V}} \end{bmatrix} \frac{\partial \vec{V}}{\partial \vec{X}} = - \begin{bmatrix} \frac{\partial p}{\partial \vec{V}} \\ \frac{\partial l}{\partial \vec{V}} \end{bmatrix} \tag{6}$$

Using the above equations and given a nominal guess for the position, the updated position estimate becomes a standard weighted least-squares problem:

$$H = \begin{bmatrix} \frac{\partial p_i}{\partial \vec{X}} \\ \frac{\partial l_i}{\partial \vec{X}} \end{bmatrix}, \quad i = 1, N \tag{7}$$

$$Y = [\Delta p_i \ \Delta l_i]^T, \quad i = 1, N \tag{8}$$

$$\hat{X} = [H^T W H]^{-1} H^T W Y \tag{9}$$

where

$\hat{X}$ = estimated correction to the position

Δp, Al = the observed minus computed pixel and line residuals

W = the weighting matrix

N = the number of limb observations.

The term in brat.kct,s in (9) is the formal covariance matrix of the estimate which will be used later in the orbit determination filter. The weighting matrix, W, is the inverse square of the uncertainties Of each pixel and line residual pair. How this value is determined will be described in the next subsection.

3.3 Limb Extraction and Correlation

The last piece of the image processing system is the actual limb correlation technique which matches the predicted limb locations with the true locations seen by the camera. This process must be fairly robust, for if an incorrect match is found, the solution obtained from the fill er above will be in error. Historically, this correlation was simplified by the fact that the initial matching of the limbs is done by a human, thus, avoiding gross misidentifications. To automate this process to a certain degree, the procedure used is as follows.

First, compute the center-of-brightness (COB) in the modeled and observed pictures. Figure 2 shows the location of the COB, marked by an 'o', computed for a modeled scene (Fig. 2a) and the "true", or observed scene (Fig. 2b). Depending on the phase, the COB will be offset from the center-of-figure (COF, used here as being the same as the center-of-mass of the asteroid, shown in Figure 2 as an 'x'). The value of the offset computed from the model will be used to guess the COF of the observed scene. The limb scans will be taken from the respective COFs of the modeled and observed scene.

The next step is to find candidate limbs to correlate. In order to get, the maximimal geometric information!, it is desirable to obtain limbs as far separated as possible. However, unless the asteroid is seen at full phase (an unlikely event, given most mission constraints), only limbs from the lit side of the object will be usable. Of all locations on the lit side, an obvious first choice is the limb lying along the sun direction in the camera frame. The sun angle in the camera frame, ϕ, can be found from

$$\vec{S}_c = T_{IC}\vec{S} \tag{10}$$

$$\phi = tan^{-1}\left(\frac{S_{c2}}{S_{c1}}\right) \tag{11}$$

where

$\vec{S}$ = the spacecraft to sun vector in inertial coordinates, and

$\vec{S}_c$ = the spat.cc.raft to sun vector in camera coordinates.

The limb is found by sampling the brighness value (know as the DN value) along the line from the COF towards the sun direction until it falls below a certain threshold (shown as a line which starts from the COF and directed towards the sun in Figures 2a and 2b). in the same manner, radial scans from the COF arc taken to sample the limb in a 60° arc centered on ϕ for the model and a 90° arc for the observation. The angular increments at which the scans are taken are chosen to be such that the arc spanned from one limb scan to the next is close to one pixel. To ensure that this is the case, at cinch limb scan, the increment for the next scan is computed by dividing one pixel by the radius of

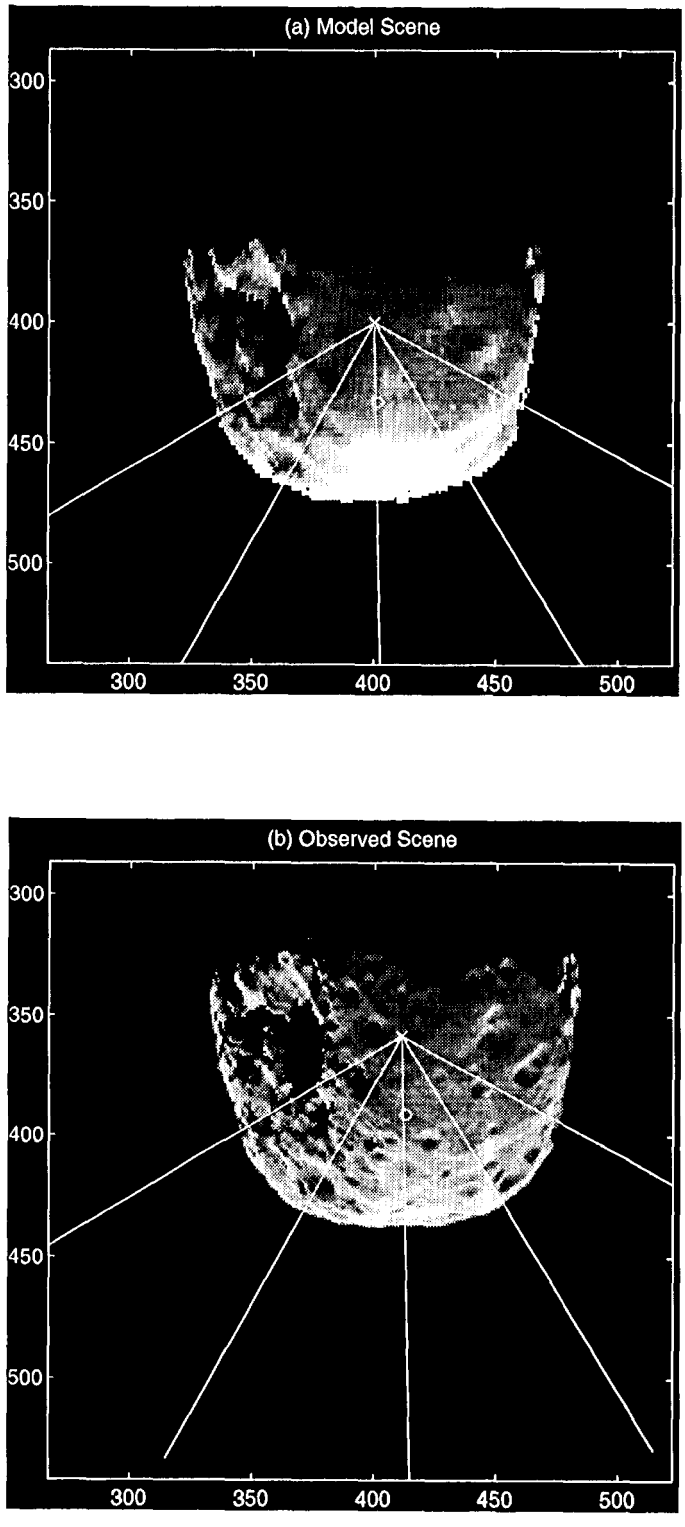

Figure 2: Center-of-brightness and Limb Scans for (a) Model Scene and (b) Observed Scene

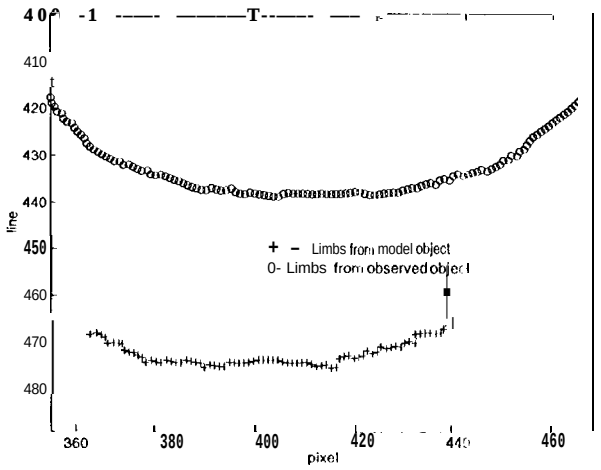

Figure 3: Limb Locations from Modeled and Observed Object for Scans Surrounding Sunline

the limb from the COF. This results in an array of about 70-80 limb samples, $\mathbf{L_m}$, for the model and 110-130 samples, $\mathbf{L_o}$, for the observation. Figure 3 plots the sampled limbs surrounding the sunline for the model and observed scene shown in Figure 2. The absolute positions of $\mathbf{L_m}$ and Lo are removed by subtracting each limb location from the first one in the set, resulting in a matrix of relative limb locations. These relative limb positions are said to be matched when

$$\| \mathbf{L_o} - (\mathbf{L_m} + \mathbf{S}) \| \tag{12}$$

is a minimum, where S is the pixel and line shift to be determined which achieves the minimum.

To find S, the smaller array of $\mathbf{L_m}$ is overlaid with a sample of $\mathbf{L_o}$ of the same length, starting from their first values, and the norm is computed. This process is repeated by incrementing the starting location of $\mathbf{L_m}$ along $\mathbf{L_o}$ until the end of LO is reached. A plot of the norm of the difference matrix is shown in Figure 4 for a candidate set of limbs. The minimum of this plot is the location along Lo at, which the limbs are best correlated. The set of absolute limb locations of these matching limbs are then recomputed, and the average difference between them provides the residual observable for the limb scan direction given by ϕ.

This procedure produces one of the Δps and Δls needed in the least-squares fit. 'J'o get other candidate limb points, the same limb scan and matching process is repeated along lines at $\pm 60°$ and $\pm 30°$ of ϕ, to get a total of five limbs (see Figure 2). When the phase becomes greater than 90°, however, the limbs near the perpendicular to the sun line become unusable, and only the $\pm 45°$ are used, for a total of three limbs. The degraded information available in these cases is reflected in the formal covariance matrix from (9), and will be accounted for in the orbit determination procedure described in the next section.

The uncertainties for each pixel and line pair of residuals is computed as follows. The minimum sigma is taken to be number of pixels corresponding to the resolution of the image. For example, for the 250 m resolution model used in the simulation seen at a distance of 85 km, the resolution of the image can be computed as roughly two pixels, based on the given wide FOV camera. 'J'he]), in order to account, for variations in the ability of the image processing to match limbs properly, the weight is taken to be the rms difference of the best matched limbs, if this value is greater than the minimum. Thus, a poorly matched limb will be assigned the weight of the rms difference, whereas a well matched

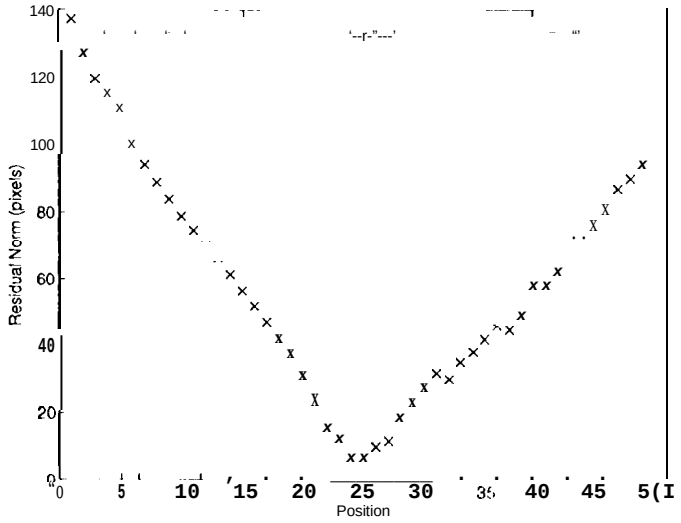

Figure 4: $||I_o$ -- $I_m||$ vs Position of I_m Along I_o

limb will be given the minimum. The uncertainties in pixel and line are assumed to be uncorrelated, so W in (9) will be a diagonal 2x2 matrix.

In practice, it was found that the procedure described usually needed two iterations to converge if the nominal position differed from the true position by more than a few degrees. The first iteration is needed to remove the angular error, from which a new model is construct which is correct in orientation but off in distance. The second iteration then removes the remaining error in range. The procedure will converge for angles even as high as 30° or so, but more iterations are needed, a time consuming process since it requires recomputation of the model scene at each iteration. In general, however, the a-priori knowledge of the orbit will be better than this, so 1 wo iterations will usually suffice.

4 Orbit Determination

The image processing procedure produces an instantaneous fix on the cartesian position of the spacecraft relative to the asteroid. It is desirable to connect these fixes together to estimate the complete state of the spacecraft. For this analysis, a simple Keplerian two-body orbit was used for the force model. Then, a standard orbit determination filter was applied which linearizes the orbit around a nominal trajectory, and computes corrections to the nominal state based on a batch least-squares fit to the observations. The estimated state included the spacecraft's position and velocity, and the gravitaional parameter, μ, of the asteroid. With the observations being the instantaneous position fixes, the observational partial matrix at a given time is simply:

$$H_{3x7} = \left[\frac{\partial(obs)}{\partial(state)}\right] = [I_{3x3} \mid 0_{3x4}] \tag{13}$$

The observations are mapped back to the epoch using the state transition matrix of the filter,

$$\tilde{H} = H\,\Phi_{7x7} \tag{14}$$

so the mapped observation partial matrix is simply the first 3x7 elements of the state transition matrix. The least-squares estimate of the corrections to the nominal state are:

$$\hat{x} = [P_0^{-1} + \tilde{H}^T W \tilde{H}]^{-1} \tilde{H}^T W Y \tag{15}$$

where

$\hat{x}$ = the corrections to the nominal state
P_0^{-1} = the a-priori covariance of the state
W = the weighting of the observations, and
Y = the vector of observations.

The weight matrix, W, is the inverse of the formal statistics from the image processing given by (9). Each batch of the filter consists of a set of at least three position fixes from the image processing filter. At set increments of time, a new estimate is computed, using the mapped covariance from the previous bate.11 estimate as the a-priori covariance matrix, P_0. 'J'he process is initialized with no a-priori knowledge, that is, P_0^{-1} = O.

5 Simulation and Results

in order to verify and test the image processing and orbit determination procedures, a simulation was performed for au orbit around an asteroid. The simulation employs a "truth" integration which is used to calculate the true geometry for building the simulated observable, and the model integration which includes errors in the initial state parameters. To add realism a nd assess the effect of systematic errors, the model of the object computed by the filter employs a different representation of the object, as wc]] as differences in the luminosity aud phase laws which describe its brightness distribution. In particular, the mode] object uses the block representation which has a surface resolution of 250 m, whereas the "truth" object uses the flat plate representation described in Ref. 1. Since the plate representation was developed primarily for visualization purposes, it incorporates interpolation between connecting plates to produce a smoother picture, while the blocks have a somewhat rougher look. The asteroid's mean radius is approximately 20 km, and its nominal gravitational parameter, computed assuming constant density, is $8.867 \times 10^{-3}\ km^3/s^2$.

The simulation was initialized with a near circular orbit with a semimajor of axis of around 85 km and an eccentricity of 0.1. The orbit is nearly perpendicular to the sunline, so the phase varies from a minimum of 80 degrees to a maximum of 97. The initial discrepancy in the spacecraft's state between the mode] and the truth trajectory is about 5 km in range and 3 degrees in angular position, and about 10 mm/s in velocity. The true μ is set to be $1.155 \times 10^{-3}\ km^3/s^2$, a discrepancy of about 30% from the model. Observations were taken at a rate of one per hour, and an orbit solution was performed after three observations, or every three hours, The minimum data weight for each limb observation was set at 2 pixels, which corresponds to the resolution at 85 km range using the camera parameters and block mode] resolution described above. Each time a solution is obtained, the estimates and covariances arc mapped forward to the next epoch and used as the a-priori solution for the next batch estimate.

The results of the simulation arc shown graphically in Figures 5 through 7. Figure 5 plots the residual between the instantaneous point position estimate made by the image processing filer and the true position in the camera x-y-z (the x and y being the pixel and line coordinate, and z is along the camera boresight) frame. Also plotted arc the 1σ error bars of the estimate. It can be seen from these plots that the filter successfully computed the correct state in just about every case, with the

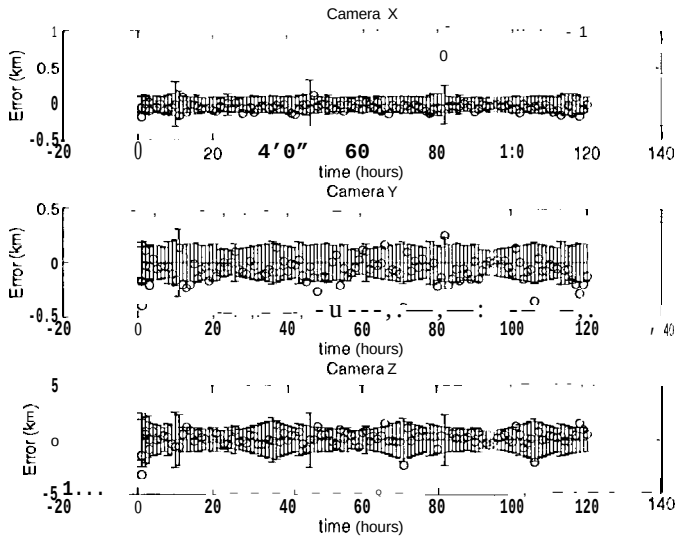

Figure 5: Position Errors (Estimated - Truth) and 10 Uncertainties from Image Processing Filter

errors being at worst slightly over 2σ in a few of the points. The formal uncertainties clearly show the procedures ability to pinpoint the spacecraft plane-of-sky position to the level of uncertainty of the model, around 200-250 m, while the range direction is much more poorly determined, around 1-2 km. It can also be seen that the range determination is more sensitive to the geometry of the scene, with the 34 hour period of the truth orbit clearly visible as a periodicity in the range uncertainty. These periodicities are also visible in the x and y directions, but not as pronounced. Finally, the statistics of the actual errors reveals a bias in the x and y estimates to be about 30 and 60 m, respectively, while the bias in range is about 115 m. This implies that the model is about 40-50" m smaller than the true object. This fact will also impact the orbit determination results.

Figure 6 plots the residual between the estimated orbit computed by the dynamic filter and true orbit, given in terms of its radial, transverse, and normal components. The plots show that after an initial period of adjustment, the estimates stabilize to very near the true values, within the 1-1.50 of their respective uncertainties. The bias in the range determination from the image processing filter is evident in here as well, but the dynamic constraints have reduced this value to an average of about 60 m. The out-of-plane componenet of the error has no biases, but the values oscillate between its 1σ values with a period equal to the orbital period. The largest bias is seen in the downtrack component, with a mean value of 100 m. In order to compensate for the larger semimajor axis, the filter also overestimates the value for μ by about 4%, shown in Figure 7.

6 Conclusions

This paper demonstrated an effective method of orbit determination using optical images as a data type. Using a wide FOV camera and a precomputed model of an asteroid, an algorithm was developed which found limbs from a location near the center of mass of the object in both the modeled and observed scene, and then correlated the limb segments to determine a best fit to the instantaneous object centered position of the observing spacecraft. The algorithm was proven by using independent models for the the truth object and modeled object, which also introduces biases into the system which are quantified in the process. These instantanous position fixes were then used as observables to a

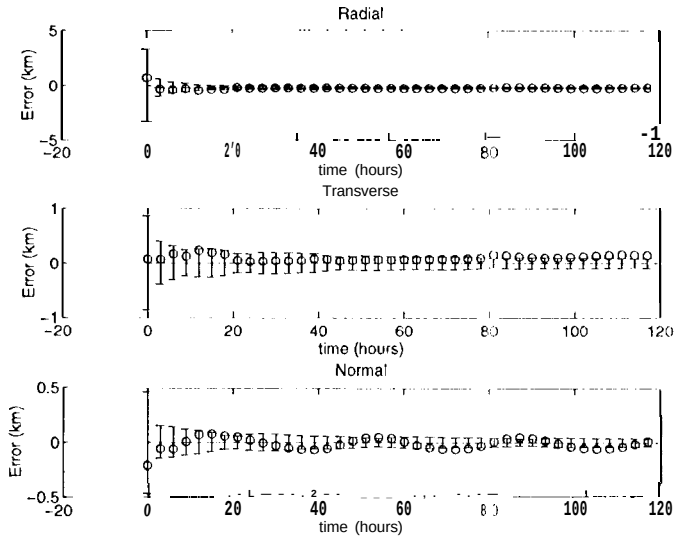

Figure 6: Position and Velocity Errors (Estimated - Truth) and 1σ Uncertainties from Orbit Determination Filter

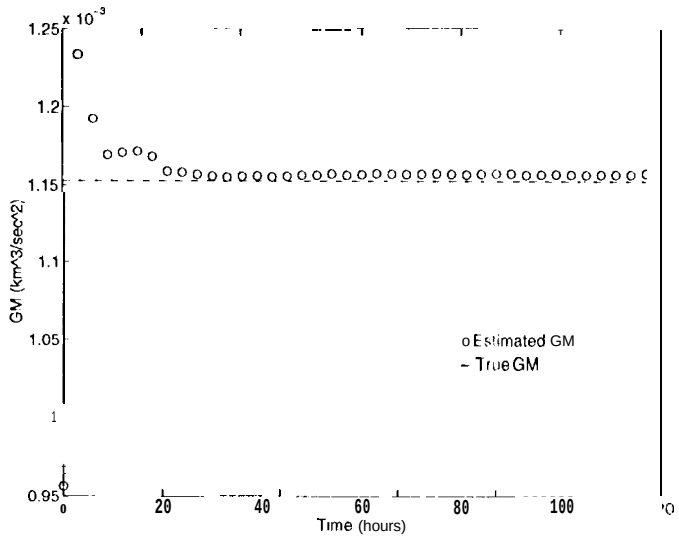

Figure 7: Estimated and True μ

dynamic filter which estimated the complete state of the spacecraft.

Although the current dynamic model uses a simple two-body force model, efforts are underway to add more realistic forces for the simulation. in particular, the effects of gravity harmonics of the asteroid, which includes offsets in the center-of-mass, and solar radiation pressure will be assessed. Finally, for complete autonomous navigation capability, orbit control in the form of autonomous maneuver computation will need to be incorporated into the simulation.

Acknowledgment

The **work** described in this paper was carried out at the Jet Propulsion laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration.

References

1. R. W. Gaskell, Personal communication.

2. D. J. Scheeres, S. J. Ostro, R. S. Hudson, and R. A. Werner, "Orbits Close to Asteroid 4769 Castalia", paper submitted to *Icarus*, April, 1995.

3. A. J . Donegan, J. A. Stuve, and J. E. Riedel, "Estimating Physical Characteristics of Astronomical Bodies from Spacecraft Images: A New Approach with Navigation Applications", J]'], Internal Document JPL-EM 314-471, September 22, 1989.

4. J. Riedel, W. Owen, J, Stuve, S. Synnott, and R. Vaughan, "Optical Navigation During the Voyager Neptune En counter", Paper presented at the AIAA/A AS Astrodynamics Conference, AIAA-90-2877, Portland Oregon, August, 1990.

5. W. M. Owen and R. M. Vaughan, "Optical Navigation Program Mathematical Models", J}'], Internal Document JPL-EM 314-513, August 9, 1991.